Paper:

Observed Body Clustering for Imitation Based on Value System

Yoshihiro Tamura*, Yasutake Takahashi, and Minoru Asada*,*****

*Graduate School of Engineering, Osaka University
2-1 Yamadaoka, Suita, Osaka 565-0871, Japan
E-mail: {yoshihiro.tamura, asada}@ams.eng.osaka-u.ac.jp
**Graduate School of Engineering, University of Fukui
3-9-1 Bunkyo, Fukui 910-8507, Japan
E-mail: yasutake@ir.his.u-fukui.ac.jp
***JST ERATO Asada Synergistic Intelligence Project
2-1 Yamadaoka, Suita, Osaka 565-0871, Japan
[Received April 15, 2010; accepted July 4, 2010]

In order to develop skills, actions, and behavior in a human symbiotic environment, a robot must learn something from behavior observation of predecessors or humans. Recently, robotic imitation methods based on many approaches have been proposed. We have proposed reinforcement learning based approaches for the imitation and investigated them under an assumption that an observer recognizes the body parts of the performer and maps them to the ones of its own. However, the assumption is not always applicable because of physical differences between the performer and the observer. In order to learn various behaviors from the observation, the robot has to cluster the observed body area of the performer on the camera image and maps the clustered parts to its own body parts based on reasonable criterion for itself and feedback the data for the imitation. This paper shows that the clustering the body area on the camera image into the body parts of its own based on the estimation of the state value in a framework of reinforcement learning as well as it imitates the observed behavior based on the state value estimation. Clustering parameters are updated based on the temporal difference error analogously so the parameters of the state value function of the behavior are updated based on the temporal difference error. The validity of the proposed method is investigated by applying it to an imitation of a dynamic throwing motion of an inverted pendulum robot and human.

Keywords: reinforcement learning, imitation, state value, clustering

1. Introduction

In order to develop skills, actions, and behavior in a human symbiotic environment, a robot often learns something from observation of predecessors or humans. Observation makes behavior learning faster and more efficient [1–3]. Recently, robotic imitation methods based on many approaches have been proposed (for example, [4, 5]). It is desirable to acquire various unfamiliar behavior with some instructions from others, for example, surrounding robots and/or humans in a real environment. Behavior learning through observation has been more important.

Reinforcement learning has been studied in learning motor skills and robot behavior acquisition in single and multiagent environments [6]. Reinforcement learning generates not only an appropriate behavior mapping from states to actions to achieve a given task but also a utility of the action, called a "state value," an estimated discounted sum of potential rewards agent receives by following a policy of the behavior. Estimation error of the state value is called "temporal difference error" (hereafter TD error) and the agent updates the state value and the behavior based on the TD error, eventually, representing its behavior based on the state value.

On the other hand, Meltzoff proposed [7] a "Like me" hypothesis that a child uses the experience of self to understand the actions, goals, and psychological states of a performer including its caregiver. From a viewpoint of reinforcement learning framework, this hypothesis suggests that the reward and state value of the performer might be estimated through observing the behavior. Takahashi et al. proposed a method understanding observed behavior based on state value estimation [8] and mutually developing observed behavior acquisition and recognition [9, 10].

The imitation based on reinforcement learning approaches has been investigated them under an assumption that the observer recognizes the body parts of the performer and maps them to the ones of the observer. However, the assumption is not always applicable because the performer physically differs from the observing robot. In order to learn various behavior from the observation of physically different performers, the robot thus must cluster the observed body area of the performer on the camera image and maps the clustered parts to its own body parts based on reasonable criteria for itself and feedback the data to behavior learning by itself.

When a child learns dynamic behavior by observing its caregiver, the caregiver intentionally demonstrate actions easy for the child to imitate – tossing the child a

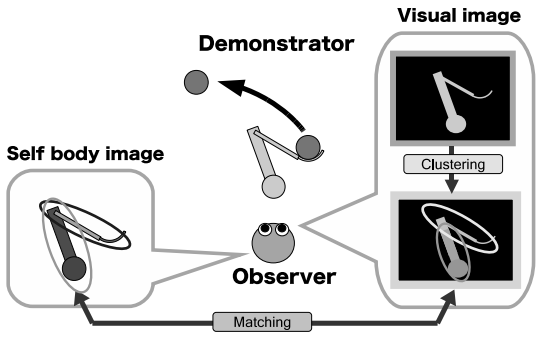


Fig. 1. Scenario of mapping an observed body image to its own: an observer watches an inverted-pendulum robot throwing a ball and maps the body parts of the performer to its own.

ball slowly rather than quickly so that the child can figure out how to catch and throw it back – an action called "motionese." The motionese can be thought that it becomes easy for the child to estimate the caregiver's error of reward, and there is an effect of taking the matching of the body part with the caregiver as a result. Nagai et al. [11, 12] state that motionese is analyzed using a saliency based attention model effective in task learning.

As our imitation method is based on reinforcement learning, especially value function, clustering the body area on the camera image based on the value system is investigated in this paper. We show a method for clustering performer's body area on the camera image for the imitation of the observed behavior based on a value system from which values are obtained in reinforcement learning. Clustering parameters are updated based on the temporal difference error (hereafter TD error: estimation error of the state value) analogously so state-value parameters of the behavior are updated based on the TD error. Preliminary investigation results by applying it to a imitation of a dynamic throwing motion of an inverted pendulum robot are shown.

2. Clustering Observed Body Area Based on TD Error

Starting with an experimental scenario, we discuss reinforcement learning, state/action value function, learning of throwing, representation of links forming a body, and clustering observed body area based on TD error.

2.1. Scenario of Experiment

As shown in **Fig. 1**, each of two inverted-pendulum robots throws a ball using an arm on the torso and has two actuators, one for the wheels and one for the torso-arm joint. Each robot has independently acquired behavior of throwing a ball and maintains a state value function based on reinforcement learning.

After learning behavior, a player demonstrates throwing. The other player, as an observer, tries to map the observed body area of the performer to its own links of the body. Parameters of the clustering of the observed body area for mapping the clusters to the observer's links of the body have to be estimated accordingly. The mapping enables the observer to understand and imitate the observed behavior based on state value function as proposed in [8, 10, 13].

2.2. State Value and TD Error

A robot is to discriminate set S of distinct world states in a world modeled on a Markov process, making stochastic transitions based on its current state and action taken by the robot based on policy π. The robot receives reward r_t at step t when it follows policy π. State value at state s_t, $V(s_t)$, the discounted sum of the reward received over time in policy π execution is calculated as follows:

$$V(s_t) = E[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots] \quad \ldots \quad (1)$$

where $0 < \gamma < 1$ is a discount rate. The robot receives a positive reward if it reaches a specified goal, otherwise zero, so, the state value increases if the robot follows an appropriate policy π. The robot updates policy π through trial and error to receive further higher positive rewards. From Eq. (1), state value V_t is derived as follows:

$$V(s_t) = E[r_{t+1}] + \gamma V(s_{t+1}). \quad \ldots \quad (2)$$

State value V_t is updated iteratively as follows:

$$V(s_t) \leftarrow V(s_t) + \alpha \Delta V(s_t) \quad \ldots \quad (3)$$

$$\Delta V(s_t) = r_{t+1} + \gamma V(s_{t+1}) - V(s_t) \quad \ldots \quad (4)$$

where $\alpha (0 < \alpha \leq 1)$ is the update ratio. $\Delta V(s_t)$ is called TD error and is used to update the parameter of estimation of the state value function and the policy. **Fig. 2(a)** shows a diagram of the state value updating procedure. For details, see [14] and [6].

2.3. Learning Throwing

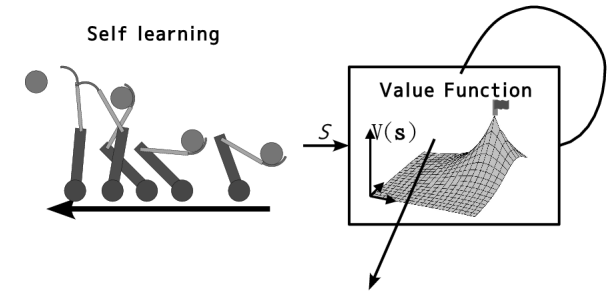
Figure 3 shows how the observer learns throwing. **Fig. 4** models the mobile inverted-pendulum robot with an arm. θ_a and θ_t are angle from the torso to the arm and angle between the torso and the direction of gravity, respectively. State variables for state space of learning throwing are θ_a, $\dot{\theta}_a$, θ_t, and $\dot{\theta}_t$.

The state value function is approximated using tile coding as a 4-dimension table. θ_a and θ_t spaces are quantized into 8 and the other state variables' spaces into 10.

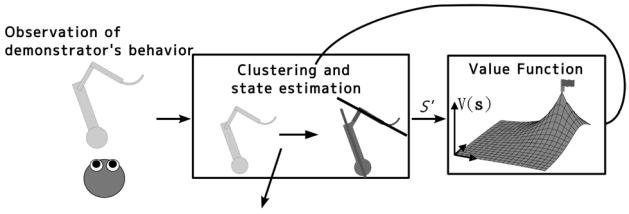
The robot learns throwing through trial and error while it receive positive reward for successfully throwing and zero-reward otherwise. State value function is updated over trials based on rewards.

2.4. Representation of Links of Observed Body

The throwing robot has one link each at the torso and arm. The observer watches the robot with a camera extracting silhouettes of the observed robot by subtracting background images. The silhouettes contain the robot torso and arm. The observer segments the silhouette into



(a)



(b)

Fig. 2. (a) Updating state values based on TD error through trial and error, (b) updating link representation parameters based on the TD error.

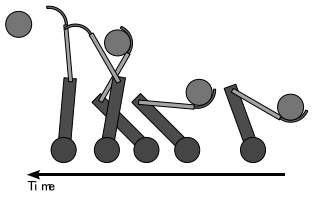


Fig. 3. Throwing motion.

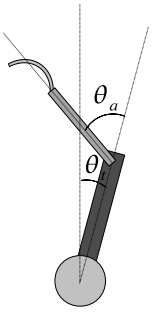


Fig. 4. Model of an inverted pendulum mobile robot with an arm.

two links. A link is modeled arbitrarily, with an ellipsoid used here for simplicity. The Mahalanobis distance simply and adequately measures clustering to links. A link has center μ and region covariance Σ. Mahalanobis distance D to the link is as follows:

$$D(x) = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)} \quad \ldots\ldots\ldots \quad (5)$$

The robot consists of torso link and arm links. The Mahalanobis distance from arbitrary point x to the torso link D_t is calculated using Eq. (5) with center μ_t and covariance

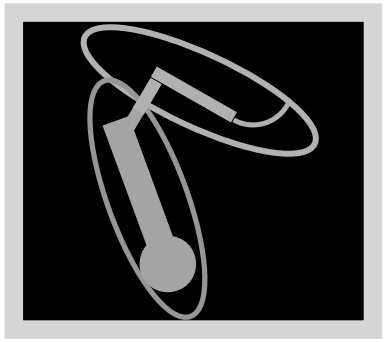


Fig. 5. Representation of links of observed body based on Mahalanobis distance.

Σ_t, with the distance to the arm link D_a with center μ_a and covariance Σ_a of the arm link. Point x in the silhouette is classified as the torso link if $D_t < D_a$, and otherwise as the arm link. **Fig. 5** represents links of observed body based on the Mahalanobis distance. The center vectors and covariance matrices of the two links are actually the clustering parameters updated based on state value function, i.e., TD errors.

After the clustering parameters and the body region on the observed image are defined as shown in **Fig. 5**, posture parameters θ_a, ξ_a, θ_t, and ξ_t, are estimated, first, by clustering pixels on the observed image with the clustering parameters based on the Mahalanobis distance the torso and arm. The postures of the arm and the torso are calculated using clustered pixels as follows:

$$\theta = \frac{1}{2} \arctan\left(\frac{2M_{11}}{M_{20} - M_{02}}\right) \quad \ldots\ldots\ldots \quad (6)$$

where M_{pq} is a region moment:

$$M_{pq} = \sum_i \sum_j (i - i_G)^p (j - j_G)^q f(i,j) \quad \ldots\ldots \quad (7)$$

(i_G, j_G) is the center and $f(i,j)$ is pixel value at point (i,j), i.e., 1 if the point on the performer silhouette, otherwise zero.

The angular velocity of torso $\dot{\theta}$ is estimated with simple numerical differentiation as follows:

$$\dot{\theta} = \left(\frac{\theta_t - \theta_{t-1}}{T}\right) \quad \ldots\ldots\ldots\ldots\ldots \quad (8)$$

where T is time step size.

2.5. Update of Link Representation Parameters Based on TD Error

During learning, the state value function is updated based on TD error as described in Sections 2.2 and 2.3 and shown in **Fig. 2(a)**. After learning, the value function is fixed, then the robot observes the other player and tries to map the observed body area of the performer to its own links of the body by updating the link representation parameters based on the TD error as shown in **Fig. 2(b)**. TD error feedback does not update state value function that is acquired beforehand. The observer watches the per-

former, estimates link postures of the performer, and updates clustering parameters μ_i and Σ_{ij} of each performer link based on the estimated TD error $\Delta\hat{V}_t$ as follows:

$$\mu_i \leftarrow \mu_i - \beta \frac{\partial |\Delta\hat{V}_t|}{\partial \mu_i} \quad \ldots\ldots\ldots\ldots (9)$$

$$\Sigma_{ij} \leftarrow \Sigma_{ij} - \beta \frac{\partial |\Delta\hat{V}_t|}{\partial \Sigma_{ij}} \quad \ldots\ldots\ldots\ldots (10)$$

where, i, j, and $\beta (0 \leq \beta \leq 1)$ are indexes of the parameter and update ratio, respectively. The condition that the TD error is zero means that the performer's link posture sequence perfectly match to the one of the throwing of the observer, thus, minimizing the TD error by updating the clustering parameters indicates that the observer maps the clustered links to its own links to represent its own throwing successfully.

Estimated TD error is calculated as follows:

$$\Delta\hat{V}_t = r_{t+1}\hat{} + \gamma V_{t+1}\hat{} - \hat{V}_t \quad \ldots\ldots\ldots\ldots (11)$$

where

$$\begin{aligned} r_{t+1}\hat{} &= r(\hat{s}_t) \\ \hat{V}_t &= V(\hat{s}_t) \\ \hat{s}_t &\leftarrow F^{hash}(x_t) \end{aligned}$$

$\hat{s}_t$, $\hat{r}_t$, and $\hat{V}_t$ are estimated state, reward, and state value at time t. F^{hash} is a hash function that maps from sensory values ${}^d x_t$ to state $s \in S$. Here, the hash function is modeled with tile coding as stated in Section 2.3.

In the experiments below, state space is quantized into a set of discrete states where the state value function is represented. When the differential of the state value is calculated, in order to avoid a function discontinuity problem, the state value is interpolated linearly and TD error of Eq. (11) is calculated with the interpolated state value. $\frac{\partial|\Delta\hat{V}_t|}{\partial\mu_i}$ and $\frac{\partial|\Delta\hat{V}_t|}{\partial\Sigma_{ij}}$ are calculated numerically as follows:

$$\frac{\partial |\Delta\hat{V}_t|}{\partial \mu_i} \leftarrow \frac{|\Delta\hat{V}_t(x_t|^{\mu_i+\delta\mu_i})| - |\Delta\hat{V}_t(x_t|^{\mu_i-\delta\mu_i})|}{2\delta\mu_i} \quad \ldots (12)$$

$$\frac{\partial |\Delta\hat{V}_t|}{\partial \Sigma_{ij}} \leftarrow \frac{|\Delta\hat{V}_t(x_t|^{\Sigma_{ij}+\delta\Sigma_{ij}})| - |\Delta\hat{V}_t(x_t|^{\Sigma_{ij}-\delta\Sigma_{ij}})|}{2\delta\Sigma_{ij}} \quad \ldots (13)$$

where $x_t|^{\mu_i+\delta\mu_i}$ and $x_t|^{\mu_i-\delta\mu_i}$ ($x_t|^{\Sigma_{ij}+\delta\Sigma_{ij}}$ and $x_t|^{\Sigma_{ij}-\delta\Sigma_{ij}}$) are estimated state value vectors $(\theta_a, \xi_a, \theta_t, \xi_t)$ of the performer while μ_i (Σ_{ij}) is increased or decreased by $\delta\mu_i$ ($\delta\Sigma_{ij}$), respectively.

The procedure of the clustering parameter learning is as follows:

1. Initialize clustering (body segmentation) arbitrarily.
2. Update clustering parameters to reduce TD error.
3. Cluster pixels in the observed body image based on the Mahalanobis distance with clustering parameters.
4. Update clustering parameters again with clustered pixels.
5. Exit if TD error converged, otherwise go to 2.

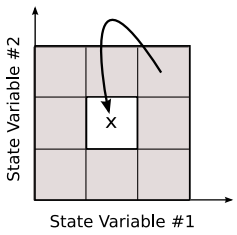


Fig. 6. Fill state value.

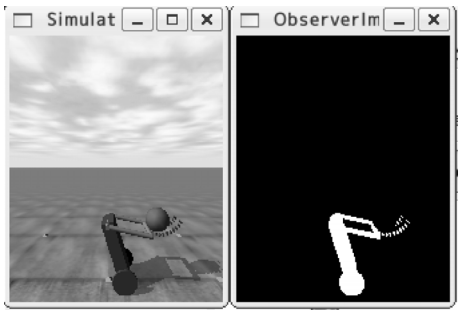


Fig. 7. View of observation.

Table 1. Physical parameters of the inverted pendulum robot.

	volume [m^3]	mass [kg]
arm	$0.02 \times 0.02 \times \pi \times 0.38$	0.75
torso	$0.1 \times 0.3 \times 0.5$	13.0
wheel	$0.1 \times 0.1 \times \pi \times 0.02$	0.5

2.6. State Value Extrapolation

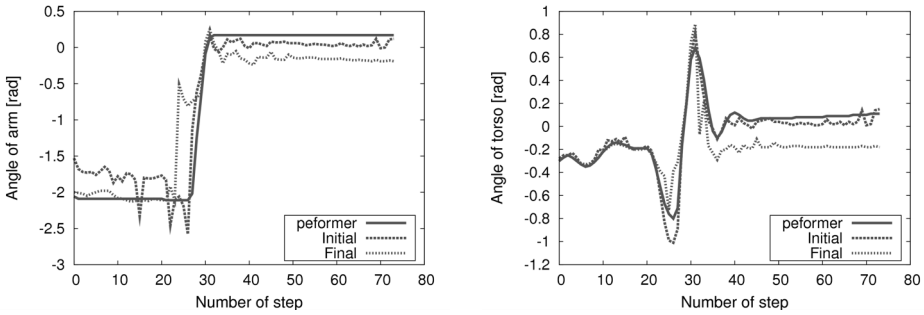
TD error from the observed trajectory may not be available because estimated angles and angular velocities of the torso and arm tends to be outside the learned state space, especially in early learning stage of classification of observed body image. Therefore, simple extrapolation of state value from the learned state to the inexperienced state is introduced as discounted value with γ of the average of the state values in adjoining learned states. The example of two-dimension state space is shown in **Fig. 6**. If the center state is inexperienced and has no state value, the discounted average state values in adjacent states is calculated as the extrapolated state value of the center state.

3. Experiment with Inverted-Pendulum Robot

Experiments use two mobile inverted-pendulum robots, each acquired throwing through trial and error based on reinforcement learning. One acts as the performer and the other as the observer. The observer's camera captures the images sequence shown at left in **Fig. 7**. It extracts performer silhouettes as shown at right in **Fig. 7**.

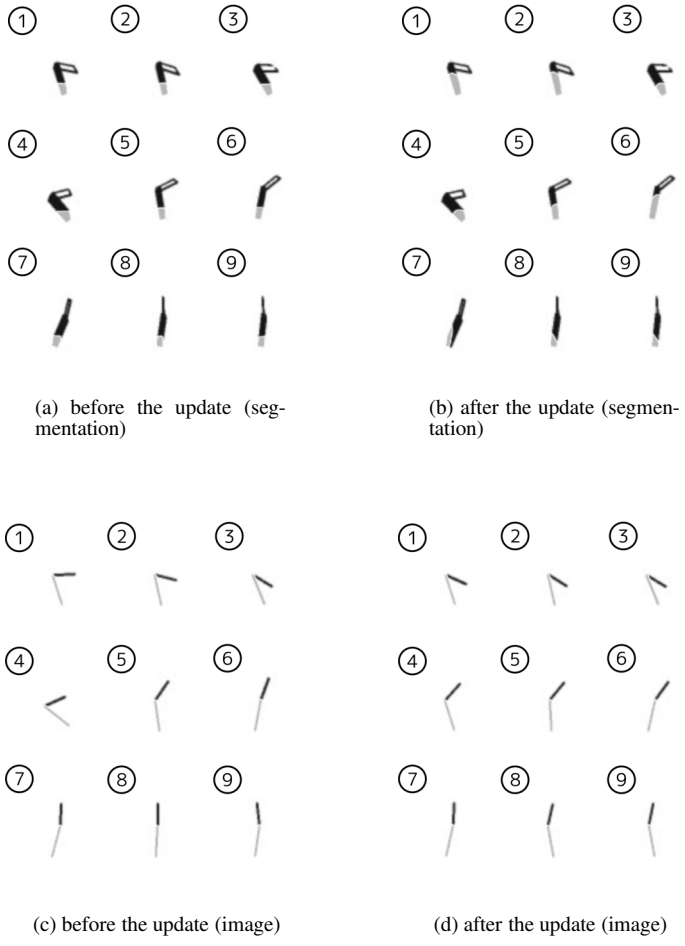
Table 1 shows physical parameters of the inverted-pendulum robot. The ball is 22 cm in diameter and weights 450 g. The dynamics is simulated by the Open Dynamics Engine (ODE).[1] The arm angle range is lim-

1. http://www.ode.org



(a) Arm (b) Torso

Fig. 8. Posture trajectory under condition 1.



(a) before the update (segmentation) (b) after the update (segmentation)

(c) before the update (image) (d) after the update (image)

Fig. 9. Body segmentation and image before/after update of clustering parameters in condition 1.

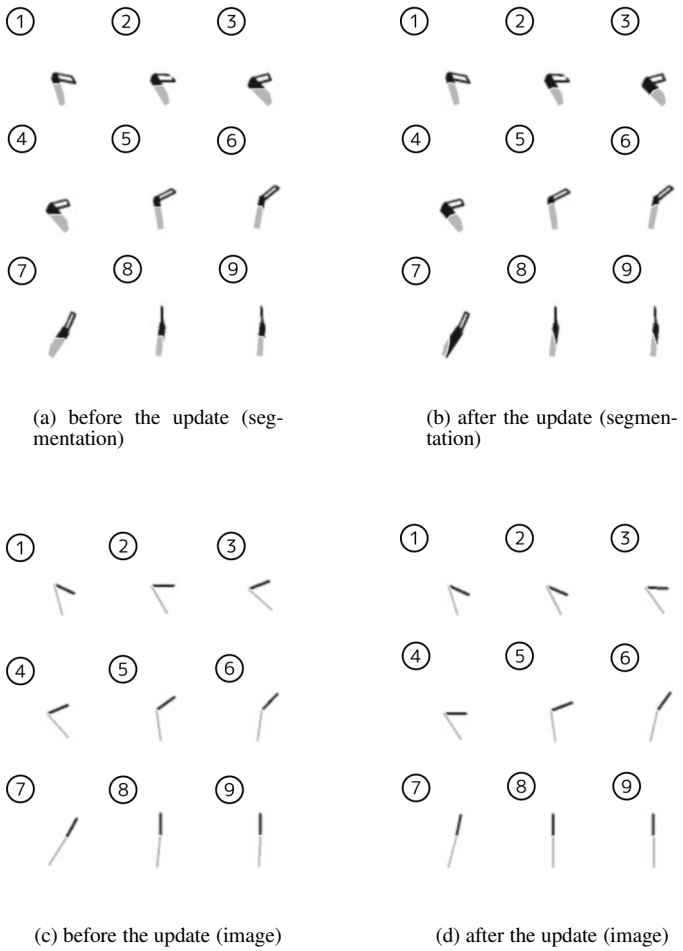
ited to $-0.84\pi < \theta_a < 0.84\pi$ rad so that the arm of the robot does not collide with the robot torso. The torso angle range is limited to $-0.5\pi < \theta_t < 0.5\pi$ rad so that the torso does not collide with the ground.

3.1. Experiments: Identical Structure and Constraints

Performer and observer have the same physical structure and constraints. Experiments confirm how body segmentation converges with different initial clustering (body segmentation) parameters. The performer and observer use identical throwing motions.

3.1.1. Condition 1

Under condition 1, the observer initializes clustering parameters with the upper three-quarters of the extracted region clustered into the arm and the rest on the torso. Clustering parameters are then updated as detailed above until TD error converges. **Fig. 8** shows estimated and ground-truth trajectories of the arm and torso posture during the throwing motion before and after clustering parameters are updated. **Figs. 9(a)** and **(b)** show the se-



(a) before the update (segmentation)

(b) after the update (segmentation)

(c) before the update (image)

(d) after the update (image)

Fig. 10. Body segmentation and image before/after update of clustering parameters under condition 2.

quence of the body segmentation before and after updating. Dark gray shows the arm and light gray the torso. The upper three-quarters of the extracted silhouette is clustered on the arm and the rest on the torso initialized before updating. At some points, after updating, half of the area is clustered on the arm and the rest on the torso. Interestingly, even though the posture trajectory is estimated reasonably well as shown in **Fig. 8**, cluster of the arm and torso differ somewhat from that expected. The torso is, for example, recognized as the smaller region and the arm is bigger than the actual. In order to clarify this result correctly, **Figs. 9(c)** and **(d)** show the sequence of body image with the sequence of the recognized posture parameters of the performer. **Figs. 9(c)** and **(d)** differ little, but, the arm angel in early throwing motion is estimated well, as confirmed in **Fig. 8(a)**. Keeping the arm folded against the torso is necessary for keeping the ball to throw. The arm inclination from number 1 to 2 in **Fig. 9(c)** is shallower than the ground-truth. The arm inclination from number 1 to 2 in **Fig. 9(d)** is close to its own throwing and early arm oscillation improves after clustering parameters updating. Therefore, arm and torso clustering appropriately converges so that clustering results explain the observed motion with its own throwing motion.

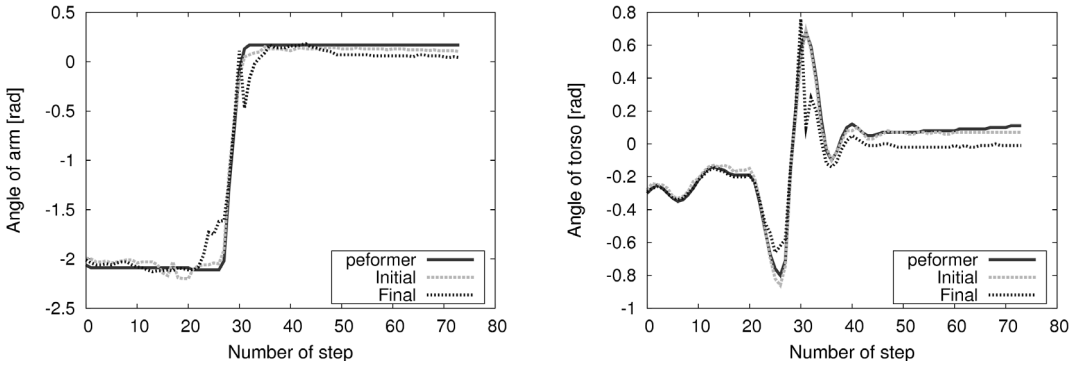
3.1.2. Condition 2

Under condition 2, the observer initializes clustering parameters with the upper half of the extracted region clustered on the arm and the rest on the body, with little difference in **Figs. 10(c)** and **(d)**. In **Fig. 11**, the estimated posture trajectory is almost the same as the performer's posture trajectory before updating but not after updating, probably due to the difference in the learning experience of the two. However, the body part matching after updating is similar to that under condition 1. Therefore, the body part matching between oneself and others is appropriate in this initial clustering parameter.

3.1.3. Condition 3

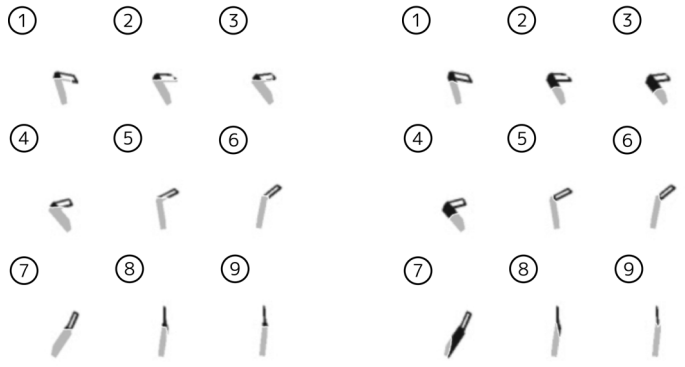
Under condition 3, the observer initializes clustering parameters in the upper quarter of the extracted region on the arm and the rest on the torso.

Figures 12(c) and **(d)** differ little. In **Fig. 13**, the estimated posture trajectory with initial clustering parameters

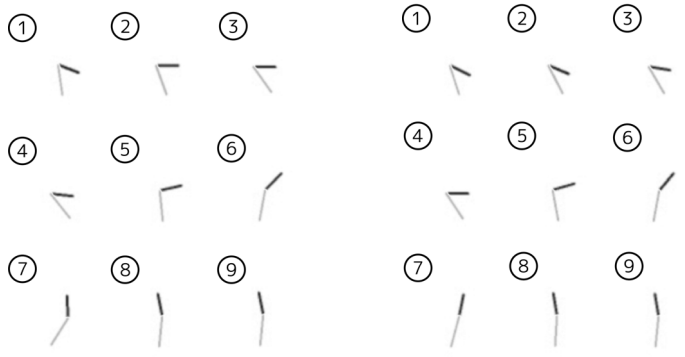


(a) Arm (b) Torso

Fig. 11. Posture trajectory.



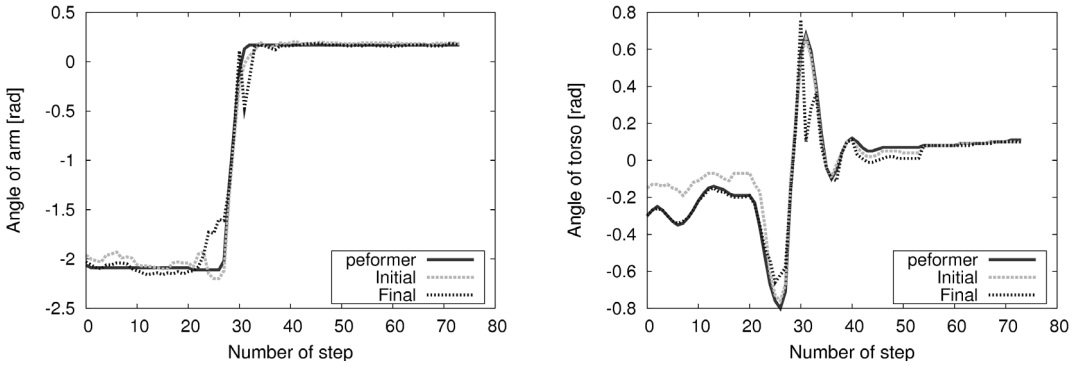
(a) before the update (segmentation) (b) after the update (segmentation)



(c) before the update (image) (d) after the update (image)

Fig. 12. Body segmentation and image before/after update of clustering parameters in condition 3.

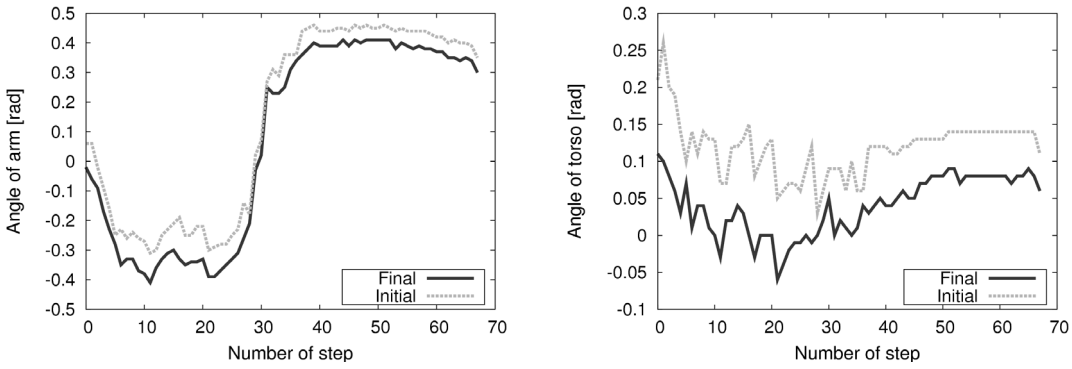
is almost the same as that of the performer. After updating parameters, estimation becomes worse, somehow. However, the result of the body part clustering after the update resembles that under conditions 1 and 2. Therefore, the body part matching between oneself and others can be regarded as converges appropriately from the initial clustering parameter.



(a) Arm (b) Torso

Fig. 13. Posture trajectory in condition 3.

Fig. 14. One sequence of human throwing motion the robot watches with a camera.



(a) Arm (b) Torso

Fig. 15. Posture trajectory in experiment of recognition of human throwing motion.

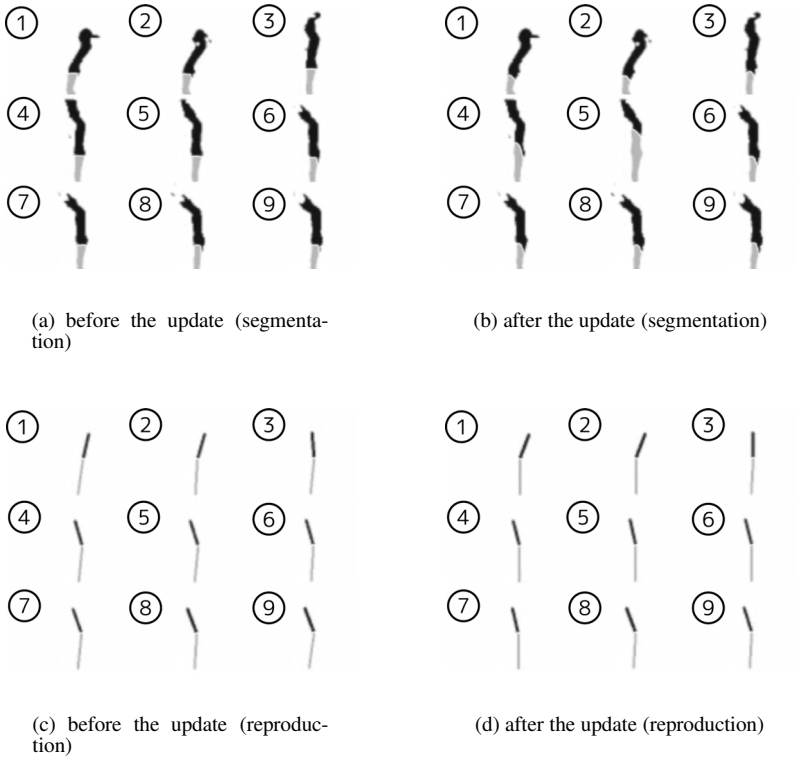
3.2. Experiments: Recognition of Body Parts of Throwing Human

In this experiment, a human performer shows throwing to the robot. The human performer is approximately 168 cm tall and weighing 54 kg. The robot watches the human performer and acquires the sequence of camera images as shown in **Fig. 14**. The observing robot extracts a silhouette of the performer by subtracting the background image.

Obviously, the robot does not have the same physical structure and constraints as the human performer. The observer initialized clustering parameters as the upper three-quarters of the extracted region clustered on the arm and the rest on the torso, then, updated clustering parameters until TD error converges.

Figure 15 shows estimated and trajectories of the arm and torso posture during throwing before and after the clustering parameters were updated. There is no ground-truth trajectory because the human body angle trajectory does not have any sense for the observing robot and the observer just recognize the human's throwing motion based on its own throwing motion.

Figures 16(a) and **(b)** show the sequence of body seg-



(a) before the update (segmentation)

(b) after the update (segmentation)

(c) before the update (reproduction)

(d) after the update (reproduction)

Fig. 16. Body segmentation and reproduction before/after update of clustering parameters for human throwing motion recognition.

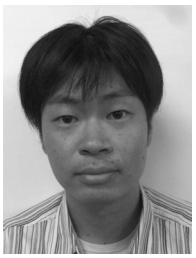
mentation before and after the clustering parameters are updated, with the arm in dark gray and the torso in light gray. **Figs. 16(c)** and **(d)** show the sequence of the body image reproduced by own body before and after the clustering parameters are updated. During the throwing motion of the observer, the torso is inclined ahead and the arm greatly raised. The inclining the torso ahead means the torso angle is less than 0 rad in **Fig. 15(b)**. In **Figs. 16(c)** and **(d)**, the arm before/after clustering parameters updating is raised from number 2 to number 4. With initial clustering parameters, the torso does not incline ahead and the torso angle exceeds than 0 from **Fig. 15(c)**. After updating, the torso inclines ahead once during throwing, indicating that the observer successfully segments the silhouette of the throwing on the image and maps them to its own torso and arm reasonably.

4. Conclusions and Future Work

This paper proposed a method for segmentation of performer's body for the imitation of the observed behavior based on a value system from which values can be obtained by reinforcement learning. The segmentation parameters are updated based on TD error, similar to how state value parameters are updated based on the TD error. Our proposed method can be easily combined with a behavior imitation method based on reinforcement learning, especially value-function-based learning. The validity of the proposed method was investigated with an imitation of dynamic throwing by a mobile inverted-pendulum robot. Results showed that clustering parameters are updated for estimation of the posture trajectory of the performer, although, the sizes and shapes of the clustered regions are different from the ones expected.

As future work, we are planning to improve our method by adding constraints on shapes and sizes of links of the body. Furthermore, extension of the proposed method is planned for simultaneous learning of state value function for the observed behavior and updating link representation parameters for observed body image in order to imitate behavior of a performer with different shapes and link configurations. The proposed method depends on the dynamics of the robot body and the motion. Limitation on the proposed method should be clarified from this viewpoint. Another point is the introduction of reinforcement learning with continuous state and action spaces, for example, [15–18]. In this paper, discrete state space is used, therefore, the proposed method requires ad hoc state value extrapolation not necessary with learning using continuous state space. Reinforcement learning handling continuous space may provide sufficient sophistication for our proposed method.

References:
[1] D. C. Bentivegna, C. G. Atkeson, and G. Chenga, "Learning tasks from observation and practice," Robotics and Autonomous Systems, Vol.47, pp. 163-169, 2004.
[2] B. Price and C. Boutilier, "Accelerating Reinforcement Learning through Implicit Imitatione," J. of Artificial Intelligence Research, Vol.19, pp. 569-629, Dec. 2003.
[3] S. D. Whitehead, "Complexity and Cooperation in Q-Learning," In Proc. Eighth Int. Workshop on Machine Learning (ML91), pp. 363-367, 1991.
[4] T. Inamura, Y. Nakamura, and I. Toshima, "Embodied Symbol Emergence based on Mimesis Theory," Int. J. of Robotics Research, Vol.23, No.4, pp. 363-377, 2004.
[5] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," 2004.
[6] J. H. Connell and S. Mahadevan, "ROBOT LEARNING," Kluwer Academic Publishers, 1993.
[7] A. N. Meltzoff, "'Like me': a foundation for social cognition," Developmental Science, Vol.10, No.1, pp. 126-134, 2007.
[8] Y. Takahashi, T. Kawamata, M. Asada, and M. Negrello, "Emulation and Behavior Understanding through Shared Values," In Proc. of the 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 3950-3955, Oct. 2007.
[9] Y. Takahashi, Y. Tamura, and M. Asada, "Behavior Development through Interaction between Acquisition and Recognition of Observed Behaviors," In Proc. of 2008 IEEE World Congress on Computational Intelligence (WCCI2008), pp. 1518-1528, June 2008.
[10] Y. Takahashi, Y. Tamura, and M. Asada, "Human Instruction Recognition and Self Behavior Acquisition Based on State Value," In Proc. of the 18th IEEE Int. Conf. on Fuzzy Systems, pp. 969-974, 2009.
[11] Y. Nagai, C. Muhl, and K. J. Rohlfing, "Toward Designing a Robot that Learns Actions from Parental Demonstrations," In Proc. of the 2008 IEEE Int. Conf. on Robotics and Automation (ICRA2008), pp. 3545-3550, 2008.
[12] Y. Nagai and K. J. Rohlfing, "Computational Analysis of Motionese Toward Scaffolding Robot Action Learning," IEEE Trans. on Autonomous Mental Development, Vol.1, No.1, pp. 44-54, 2009.
[13] Y. Takahashi, Y. Tamura, and M. Asada, "Mutual Development of Behavior Acquisition and Recognition Based on Value System," In From Animals to Animats, Vol.10 (Proc. of 10th Int. Conf. on Simulation of Adaptive Behavior, SAB 2008), pp. 291-300, July 2008.
[14] R. Sutton and A. Barto, "Reinforcement Learning: An Introduction," MIT Press, Cambridge, MA, 1998.
[15] A. Bonarini, A. Lazaric, F. Montrone, and M. Restelli, "Reinforcement Distribution in Fuzzy Q-Learning," Fuzzy Sets and Systems, Vol.160, pp. 1420-1443, 2009.
[16] K. Doya, "Temporal Difference Learning in Continuous Time and Space," In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, Advances in Neural Information Processing System, Vol.8, pp. 1073-1079, MIT Press, Cambridge, MA, 1996.
[17] T. Horiuchi, A. Fujino, O. Katai, and T. Sawaragi, "Fuzzy Interpolation-Based Q-Learning with Continuous Inputs and Outputs," Trans. of the Society of Instrument and Control Engineers, Vol.35, No.2, pp. 271-279, 1999.
[18] Y. Takahashi, M. Takeda, and M. Asada, "Improvement Continuous Valued Q-learning and its Application to Vision Guided Behavior Acquisition," In The Fourth Int. Workshop on RoboCup, pp. 255-260, 2000.



Name:
Yoshihiro Tamura

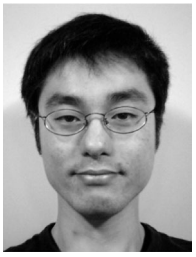
Affiliation:
Graduate School of Engineering, Osaka University

Address:
2-1 Yamadaoka, Suita, Osaka 565-0871, Japan

Brief Biographical History:
2004-2008 Faculty of Engineering, Osaka University
2008-2010 Graduate School of Engineering, Osaka University

Main Works:
• Y. Takahashi, Y. Tamura, and M. Asada, "Mutual Development of Behavior Acquisition and Recognition Based on Value System," From Animals to Animats, Vol.10 (Proc. of 10th Int. Conf. on Simulation of Adaptive Behavior, SAB 2008), pp. 291-300, 2008.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)



Name:
Yasutake Takahashi

Affiliation:
Graduate School of Engineering, University of Fukui

Address:
3-9-1 Bunkyo, Fukui 910-8507, Japan

Brief Biographical History:
2000-2009 Assistant Professor of Department of Adaptive Machine Systems, Graduate School of Engineering, Osaka University
2003-2009 Member of Exec Committee for RoboCup middle size league
2006-2007 Visiting Researcher at the Fraunhofer IAIS
2009- Senior Assistant Professor of Department of Human and Artificial Intelligent Systems, Graduate School of Engineering, University of Fukui

Main Works:
• S. Takamuku, Y. Takahashi, and M. Asada, "Lexicon acquisition based on object-oriented behavior learning," Advanced Robotics, Vol.20, No.10, pp. 1127-1145, 2006.
• Y. Takahashi and M. Asada, "Modular Learning Systems for Behavior Acquisition in Multi-Agent Environment," Cornelius Weber, Mark Elshaw and Norbert Michael Mayer (Ed.), Reinforcement Learning, Theory and Applications, Chapter 12, pp. 225-238, I-TECH Education and Publishing, 2008.
• Y. Takahashi, K. Noma, and M. Asada, "Efficient Behavior Learning based on State Value Estimation of self and Others," Advanced Robotics, Vol.22, No.12, pp. 1379-1395, 2008.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japanese Society for Artificial Intelligence (JSAI)
• The Institute of Electrical and Electronics Engineers (IEEE)

Name:
Minoru Asada

Affiliation:
Graduate School of Engineering, Osaka University
JST ERATO Asada Synergistic Intelligence Project

Address:
2-1 Yamadaoka, Suita, Osaka 565-0871, Japan

Brief Biographical History:
1995-1997 Professor of Mechanical Engineering for Computer-Controlled Machinery, Faculty of Engineering, Osaka University
1997- Professor of the Department of Adaptive Machine Systems, Graduate School of Engineering, Osaka University
1986-1987 Visiting Researcher, Center for Automation Research, University of Maryland, USA
2005-2011 Leader of JST ERATO Asada Synergistic Intelligence Project

Main Works:
• A. Watanabe, M. Ogino, and M. Asada, "Mapping Facial Expression to Internal States Based on Intuitive Parenting," J. of Robotics and Mechatronics, Vol.19, No.3, pp. 315-323, 2007.
• N. M. Mayer and M. Asada, "RoboCup Humanoid Challenge," Int. J. of Humanoid Robotics, Vol.5, No.3, pp. 335-351, 2008.
• H. Sumioka, Y, Yoshikawa, and M. Asada, "Reproducing Interaction Contingency Toward Open-Ended Development of Social Actions: Case Study on Joint Attention," IEEE Trans. on Autonomous Mental Development, Vol.2, No.1, pp. 40-50, 2010.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• Information Processing Society of Japan (IPSJ)
• The Japanese Society for Artificial Intelligence (JSAI)
• The Japan Society of Mechanical Engineers (JSME)
• The Society of Instrument and Control Engineers (SICE)
• The Institute of Systems, Control and Information Engineers (ISCIE)
• The Institute of Electrical and Electronics Engineers (IEEE)